

Programski prevodioci

1. (10 poena) Naredba za deklaraciju promenljivih u programu je definisana sledećom gramatikom:

$Dec \rightarrow \text{TYPE } VarList ;$

$VarList \rightarrow Var \mid VarList , Var$

$Var \rightarrow ID \mid ID = CONST$

- a) Transformisati datu gramatiku u LL(1) gramatiku i dokazati da tako transformisana gramatika jeste LL(1) gramatika.
- b) Kreirati LR sintaksnu tabelu date gramatike.
2. (16 poena) Kreirati *cup* specifikaciju za generisanje sintaksnog i semantičkog analizatora naredbe za deklarisanje promenljivih koja je definisana gramatikom datom u zadatku 1. Za generisanje potrebnog leksičkog analizatora kreirati odgovarajuću *flex* specifikaciju. Terminalni simboli su definisani na sledeći način:
- **TYPE** – ime tipa: **char, int, double**;
 - **ID** – niz slova, cifara, znakova '\$' i '_' u kojem prvi znak ne može biti cifra;
 - **CONST** – može biti tipa int (niz dekadnih cifara), tipa double (eksponencijalni zapis realnog broja u jeziku C) i tipa char (znakovni tip u jeziku C).

Semantička pravila jezika su:

- Promenljiva zadatak imena može biti deklarirana samo jednom.
- Ukoliko se u deklaraciji postavlja i inicijalna vrednost promenljive, mora da se slaže tip promenljive i konstante koja joj se dodeljuje.

NAPOMENA: Sve akcije u vezi sa proverom semantičke ispravnosti pisati samo uz ispravne zapise smena (smene bez **error** simbola), a smene sa **error** simbolima koristiti samo za oporavak od sintaksnih grešaka.

3. (6 poena) Definirati klasu za predstavljanje „Select“ naredbe u apstraktnom sintaksnom stablu. „Select“ naredba je definisana sledećom gramatikom:

$SelectStatement \rightarrow \text{select } AlternativeList \text{ end}$

$AlternativeList \rightarrow AlternativeList \text{ or } Alternative \mid Alternative$

$Alternative \rightarrow \text{when } Expression \Rightarrow Statement$

Definirati klasu za predstavljanje „Select“ naredbe u AST-u. Definirati međukodove niskog nivoa za izvršavanje ovako definisane naredbe i u klasi implementirati metode za generisanje međukoda. Značenje ovakve strukture: Izvršiće se prva alternativa čiji je uslov zadovoljen (tj. čiji izraz ima vrednost različitu od nule). Ukoliko nijedan od navedenih uslova nije zadovoljen, izvršenje programa se nastavlja od prve naredbe navedene nakon ove strukture.

4. (8 poena) Definirati 8086 kod i izgled aktivacionog sloga za funkciju *test_score*. Za predstavljanje podataka tipa int kao i memorijskih adresa koriste se 2 bajta. Pretpostaviti da se rezultat funkcije *test_score* smešta u registar CX, a rezultat funkcije *execute_query* na stek.

```
struct question {
    int score, id;
    struct question* next;
};

int test_score(struct question* q, int sum) {
    int condition;
    condition = execute_query(q->id);
    if (condition == 1)
        sum = sum + q->score;
    if(q->next != 0)
        return test_score(q->next, sum);
    else return sum;
}
```